

Team Ron
Project Title: MatchMentor
Members: Nallos, Hezron A.

Project Description

The project aims to develop an interactive platform to enhance student engagement and support within educational institutions. The platform will facilitate communication between students and faculty, provide access to academic resources, and offer personalized support services. The intended user population includes high school and college students, teachers, and academic advisors.

Requirements Summary

- User Authentication: Secure login for students and faculty.
- Communication Tools: Chat and video call features.
- Resource Access: Centralized repository for academic resources.
- Personalized Support: AI-driven personalized academic advising.
- Feedback Mechanism: Tools for gathering and analyzing student feedback.

Design Space

The design space of potential interfaces for our system encompasses a wide range of possibilities, reflecting different approaches to solving the problem of enhancing student engagement and support. This section explores various dimensions of the design space, including the complexity of requirements, trade-offs, and task support.

What requirements may be difficult to realize?

AI-Driven Personalized Support:

Implementing advanced AI algorithms to provide personalized academic advice tailored to individual students' needs is a significant challenge. This requires sophisticated data analysis, machine learning models, and ensuring data privacy and security.

Seamless Communication Tools:

Developing robust and reliable real-time communication tools that can handle high traffic and ensure smooth user experience across different devices can be technically complex.

Offline Access:

Enabling offline access to resources and support features in a mobile app requires sophisticated data synchronization mechanisms and local storage solutions, which add complexity to the design and development process.

What are some tradeoffs that you should or did explore?

- **Complexity vs. User-Friendliness:**
 - Balancing rich features with ease of use is crucial. Adding more features can enhance functionality but may make the interface more complicated and harder to navigate. Conversely, simplifying the interface may improve usability but at the cost of reduced functionality.
- **Performance vs. Aesthetics:**
 - High-quality graphics and animations can make the interface visually appealing but may affect performance, especially on lower-end devices. We need to find a balance between a visually appealing design and maintaining fast, responsive performance.
- **Integration vs. Standalone Features:**
 - Integrating the platform with existing tools and systems like LMS and social media can enhance usability but may introduce dependencies and complexities in maintaining compatibility. A standalone approach simplifies development but may limit the platform's overall functionality.

Which tasks will be easiest to support? Which are the hardest?

Easiest Tasks to Support:

- **User Authentication:**
 - Implementing secure login mechanisms using widely accepted practices like OAuth or single sign-on.
- **Resource Access:**
 - Providing a centralized repository for academic resources with straightforward navigation and search capabilities.
- **Feedback Mechanism:**
 - Creating tools for gathering and analyzing student feedback through surveys and polls is relatively simple using existing frameworks.

Hardest Tasks to Support:

- **AI-Driven Personalized Support:**
 - Developing and maintaining advanced AI models that provide accurate and useful advice tailored to individual students' academic progress.
- **Real-Time Communication Tools:**
 - Ensuring reliable, high-quality real-time communication (chat, video calls) that works seamlessly across different devices and network conditions.
- **Data Synchronization for Offline Access:**
 - Implementing efficient and reliable data synchronization mechanisms to ensure that users can access and update information offline and sync changes when reconnected.

This exploration of the design space helps us understand the scope of possibilities and constraints, guiding us in developing diverse and innovative design alternatives that address the core problem effectively.

Design Summary

- **Alternative 1: Mobile App Focus**
 - Not pursued
 - Limited screen space for detailed resources.
- **Alternative 2: System-based Platform**
 - Pursued
 - More screen real estate and easier integration with other web tools.
- **Alternative 3: Hybrid Approach**
 - Pursued
 - Combines the flexibility of system platforms with the accessibility of mobile apps.

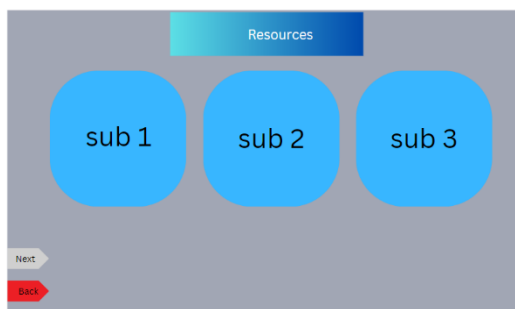
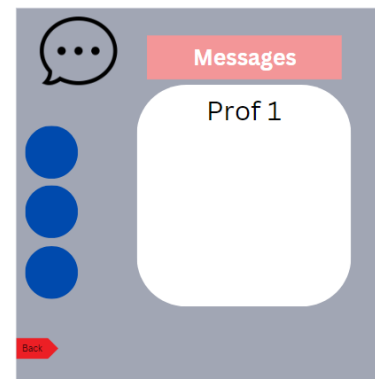
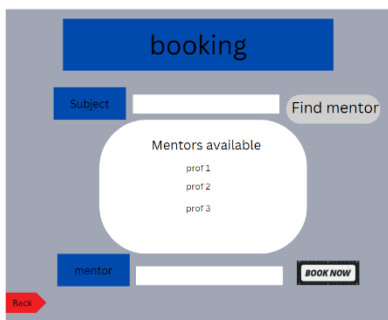
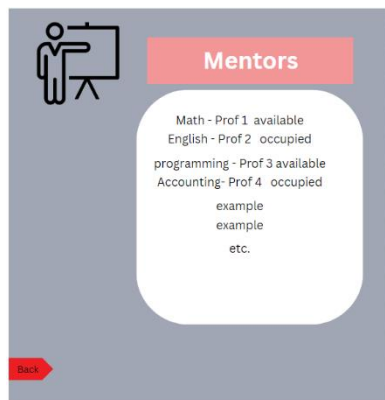
The Designs

Design 1: System-based Platform

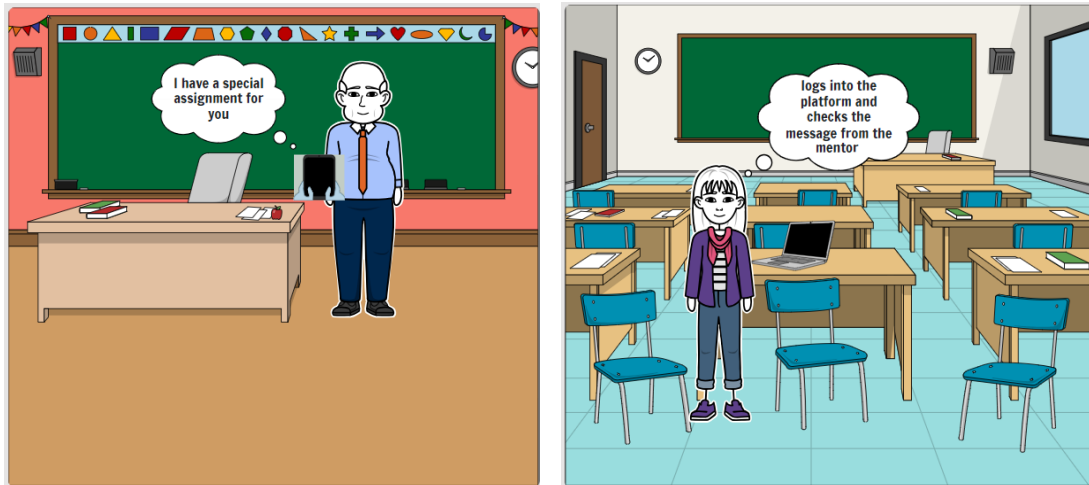
Overview:

A web-based platform providing centralized access to resources, communication tools, and personalized support.

GUI Illustrations:



Storyboard:



User Perspective Scene:

Jane, a college student, logs into the platform, checks her messages from her advisor, and accesses her course materials.

Assessment:

- Advantages:
 - Easy access, robust features.
- Disadvantages:
 - Requires internet connectivity.
- User Feedback:
 - Positive feedback on the layout and ease of use. Concerns about internet dependency.

Design 2: Mobile App

Overview:

A mobile app designed for on-the-go access to communication and support tools.

Assessment:

- Advantages:
 - Portability, instant notifications.
- Disadvantages:
 - Limited screen space.
- User Feedback:
 - Users appreciated the portability; some found the navigation challenging on smaller screens.

Design 3: Hybrid Approach

Overview:

A hybrid approach combining web and mobile interfaces for flexibility.

Assessment:

- Advantages
 - Flexibility, cross-platform continuity.
- Disadvantages
 - Requires synchronization.
- User Feedback
 - Positive feedback on the flexibility but some technical issues with synchronization.

Requirements Changes

- Changes:
 - Added requirement for offline access in the mobile app.
- Reason:
 - Feedback from users about internet dependency.